

REMARKS

No claims have been cancelled or added. Claims 9, 15 and 21 have been amended solely to correct typographical errors. Claim 1 has been amended to further clarify its subject matter. Claims 1-28 are currently pending in the application. In view of the following remarks, Applicant respectfully requests withdrawal of the rejections and forwarding of the application onto issuance.

The § 102 Rejections

Claims 1-28 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Pub. No. 2004/0096110 A1 to Yogeshwar et al ("Yogeshwar").

Applicant's Disclosure

Applicant disclosure pertains to methods and systems to enable uncompressed source data, corresponding to previously-compressed data, to be manipulated or otherwise modified, prior to being rendered by a rendering application. For example, audio data can be modified to include additional audio content, and/or video data can be modified to include additional video content. Accordingly, when the modified or manipulated source data is rendered by the rendering application, it can contain additional information that was not part of the previously-compressed data. The methods and systems can be employed in a wide variety of different areas such as advertising and content protection to name just a few.

In one embodiment, Applicant discloses that a file containing compressed data is processed in such a way as to associate it with a decoder that does *not* correspond to the encoder that originally compressed the file's source data. The

1 file is then processed by the decoder in such a way as to uncompress the
2 compressed data, manipulate or modify the uncompressed data in some desirable
3 way, and then provide the manipulated or modified data to a rendering application
4 for rendering.

5 In one embodiment, Applicant discloses that a new or second decoder is
6 provided on the client end and registered with the operating system. This second
7 decoder is associated with the new ID tag so that when the client attempts to
8 render the file, the new decoder is used in the decoding process. As an example,
9 consider Fig. 6.

10 There, a rendering application 600 is engaged by the user in an attempt to
11 render or otherwise play file 404a. In the usual manner, the rendering application
12 600 retrieves or otherwise receives file 404a and searches for the file's ID Tag.
13 Instead of finding the original ID Tag that is associated with the encoding encoder
14 (e.g. encoder 402 (Fig. 4)), the rendering application locates the New ID Tag and
15 performs a query on the New ID Tag. When the query is executed, instead of
16 returning a reference for the decoder associated with the encoding encoder, the
17 query returns a reference that can be used to load a new decoder 602. The new
18 decoder 602 functions to enable the compressed file to be uncompressed so that
19 the uncompressed source data can be manipulated in some way.

20 Fig. 7 shows but one example of new decoder 602 and the processing that
21 takes place using the new decoder. Here, decoder 602 receives file 404a which
22 may or may not contain the New ID Tag. In this example, decoder 602 is a
23 "wrapper" around an original decoder 403 that is associated with the encoding
24 encoder (i.e. encoder 402). New decoder 602 provides the compressed data of
25 file 404a to the original decoder 403 so that the original decoder can uncompress

1 the compressed data to provide uncompressed source data 400. Next, a
 2 modification module 700 receives the uncompressed source data or otherwise
 3 operates on the uncompressed source data to provide modified uncompressed data
 4 702. It should be appreciated and understood that although modification module
 5 700 is illustrated as comprising part of the new decoder 602, such need not
 6 necessarily be the case. For example, the modification module can comprise a
 7 separate software component that is called by the new decoder 602 to modify the
 8 uncompressed source data. If this is the case, then once the modification module
 9 modifies the source data, it can call the new decoder 602 and provide the
 10 modified uncompressed data 702 back to the new decoder. This modified data
 11 can now be provided to a media player or rendering application for rendering.

12 13 Claims 1-8

14 As amended, claim 1 recites a method comprising [amended language
 15 appears in bold italics]:

- 16 • providing compressed data that has been compressed using a first
 17 encoder having an associated first decoder that can be used to
 18 uncompress the compressed data;
- 19 • providing the compressed data to at least one second decoder that is
 20 different from the first decoder *and which is involved in actually
 21 causing the compressed data to be uncompressed*;
- 22 • uncompressing the compressed data to provide uncompressed data;
 23 and
- 24 • operating on the uncompressed data to provide modified
 25 uncompressed data.

23 Applicant has amended this claim to clarify that the compressed data is
 24 provided to at least one second decoder that is different from the first decoder *and*
 25 *which is involved in actually causing the compressed data to be uncompressed.*

1 In making out the rejection of this claim, the Office argues that Yogeshwar
2 teaches the subject matter of this claim. In response to Applicant's request for
3 clarification from the Office, the Office cited to paragraph 143 to support its
4 argument. The cited excerpt, discussing Yogeshwar's Fig 6, is reproduced below
5 [emphasis by Office]:

6 [0143] The image processor 604 performs one or more image
7 filtering operations or other image processing operations on the
8 decoded image data under the direction of the control module 320
9 prior to the processed image data being supplied to encoder circuitry
10 606. Encoder circuitry 606 includes a *plurality of M encoders* 608,
11 610 *each supporting a different encoding format*. An encoder
12 control module 612 interfaces with the AVARS control module 320
13 to determine which encoders 608, 610 should be enabled based on
14 the user specified output format(s) and encoding parameters such as
15 resolution and output data constraints. When multiple encoded
16 output formats are specified, the encoder 608, 610 supporting each
17 of the specified formats will be enabled. In the case where the user
18 selected output is similar to the retrieved IAF file format only *partial*
19 *decoding and re-encoding* may be required to convert the retrieved
20 file into the user specified output format. Accordingly, some of the
21 encoders 608, 610 may be partial as opposed to full video encoders.

22 Applicant will focus its attention on the *decoding* aspect of the cited
23 excerpt. Yogeshwar's "multi-format decoder" shown in Fig 6 performs the partial
24 decoding referenced above. The multi-format decoder in Fig 6 apparently has the
25 same structure as the multi-format decoder in Fig 5. Additional information
regarding the multi-format decoder in Fig 5 can be found in paragraph 107,
reproduced below [emphasis added]:

26 [0107] The multi-format decoder 519 includes a plurality of full or
27 partial decoder circuits 520, 522 each of which is designed to decode
28 data which has been encoded according to a different encoding
29 scheme. *For example, decoder 520 may be an MPEG decoder,*
30 *decoder 522 a JPEG decoder,* with other decoders supporting yet

1 other encoding formats. While the received compressed input is
2 supplied to each of the decoders 520, 522, the data analysis/decoder
3 control module, working in conjunction with control module 320,
4 determines the type of decoding to be used with any particular set of
5 input data. *With the type of decoding to be performed determined,*
6 *the control module 521 enables via a control signal the appropriate*
7 *one of the decoders 520, 522 so that one of the plurality of decoders*
8 *is used to generate the decoded digital A/V data supplied to the*
9 *scene analysis module 504 when encoded input data is being*
10 *processed.*

11 According to the excerpt reproduced just above, if compressed input
12 was encoded by an MPEG encoder, for example, Yogeshwar's system
13 provides the compressed data directly to an associated MPEG decoder.
14 Likewise, if the compressed input was encoded by a JPEG encoder,
15 Yogeshwar's system provides the compressed data directly to an associated
16 JPEG decoder. In the language of Applicant's claim 1, only Yogeshwar's
17 *first decoder* associated with the encoder is involved in actually causing the
18 compressed data to be uncompressed. As such, Yogeshwar teaches *directly*
19 *away* from the subject matter of this claim. Applicant has reviewed the
20 entire Yogeshwar reference and can find no teaching anywhere of
21 providing compressed data to at least one second decoder that is different
22 from the first decoder *and which is involved in actually causing the*
23 *compressed data to be uncompressed.*

24 Accordingly, for at least these reasons, this claim is allowable.

25 **Claims 2-8** depend from claim 1 and, as such, are allowable as depending
from an allowable base claim. These claims are also allowable for their own
recited features which, in combination with those recited in claim 1, are neither
shown nor suggested by the references as cited and applied by the Office.

Claims 9-14

Claim 9 recites a method comprising [emphasis added]:

- providing a compressed file that has been compressed using a first encoder having an associated first decoder that can be used to uncompress the compressed file, the compressed file comprising at least one ID tag that is associated with a *second decoder* that is *different from the first decoder* and that serves as a *wrapper for the first decoder*;
- searching for said at least one ID tag to identify the second decoder;
- providing the compressed file to the second decoder so that the compressed file can be uncompressed;
- *using the second decoder, providing the compressed file to the first decoder*;
- uncompressing the compressed file using the first decoder to provide an uncompressed file;
- providing the uncompressed file to a modification module;
- modifying the uncompressed file using the modification module to provide a modified uncompressed file;
- providing the modified uncompressed file to the *second decoder*;
- *using the second decoder, providing the modified uncompressed file to a rendering application*; and
- rendering the modified uncompressed file on a client device using the rendering application.

In making out the rejection of this claim, the Office again argues that Yogeshwar teaches the subject matter of this claim. In response to Applicant's request for clarification from the Office, the Office cited to paragraph 143 to support its argument. The cited excerpt, discussing Yogeshwar's Fig 6, is reproduced above.

Applicant will again focus its attention on the *decoding* aspect of the cited excerpt. Yogeshwar's "multi-format decoder" shown in Fig 6 performs the partial decoding referenced in the excerpt. The multi-format decoder in Fig 6 apparently

1 has the same structure as the multi-format decoder in Fig 5. Additional
2 information regarding the multi-format decoder in Fig 5 can be found in
3 paragraph 107, also reproduced above.

4 According to the excerpt disclosing the multi-format decoder, if the
5 compressed input was encoded by an MPEG encoder, for example, Yogeshwar's
6 system provides the compressed data directly to an associated MPEG decoder.
7 Likewise, if the compressed input was encoded by a JPEG encoder, Yogeshwar's
8 system provides the compressed data directly to an associated JPEG decoder. In
9 the language of Applicant's claim 9, Yogeshwar does not teach a *second decoder*
10 that is *different from the first decoder* and that serves as a *wrapper for the first*
11 *decoder*. As such, Yogeshwar cannot possibly teach using a *second decoder* to
12 provide the compressed file to the *first decoder*. Rather, Yogeshwar uses its "*data*
13 *analysis/decoder control module*" to provide the compressed file to the first
14 decoder. Applicant has reviewed the entire Yogeshwar reference and can find no
15 teaching anywhere of a second decoder *that is different from the first decoder*
16 and that serves as a *wrapper for the first decoder*. Nor can Applicant find any
17 teaching of using a *second decoder* to provide the compressed file to the first
18 decoder; providing a modified uncompressed file to a *second decoder*; and, using
19 the *second decoder*, providing a modified uncompressed file to a rendering
20 application.

21 Accordingly, for at least these reasons, this claim is allowable.

22 Claims 10-14 depend from claim 9 and, as such, are allowable as
23 depending from an allowable base claim. These claims are also allowable for
24 their own recited features which, in combination with those recited in claim 9, are
25 neither shown nor suggested by the references as cited and applied by the Office.

1
2 **Claims 15-20**

3 **Claim 15** recites a method comprising [emphasis added]:

- 4
- 5 • receiving a file comprising compressed data and information
 - 6 associated with an encoder that compressed source data
 - 7 corresponding to the compressed data, said information being
 - 8 configured for use in locating a first decoder that corresponds to the
 - 9 encoder and which can be used to uncompress the compressed data;
 - 10 • searching for the information; and
 - 11 • replacing the information with different information that is
 - 12 associated with a *second decoder* that is *different from the first*
 - 13 *decoder* and which *can be used*, at least in part, to *uncompress* the
 - 14 compressed data.

15 In making out the rejection of this claim, the Office again argues that
16 Yogeshwar teaches the subject matter of this claim. In response to Applicant's
17 request for clarification from the Office, the Office cited to paragraph 143 to
18 support its argument. The cited excerpt, discussing Yogeshwar's Fig 6, is
19 reproduced above.

20 Applicant will again focus its attention on the *decoding* aspect of the cited
21 excerpt. Yogeshwar's "multi-format decoder" shown in Fig 6 performs the partial
22 decoding referenced above. The multi-format decoder in Fig 6 apparently has the
23 same structure as the multi-format decoder in Fig 5. Additional information
24 regarding the multi-format decoder in Fig 5 can be found in paragraph 107, also
25 reproduced above.

According to the excerpt disclosing the multi-format decoder, if the
compressed input was encoded by an MPEG encoder, for example, Yogeshwar's
system can *only* use an MPEG decoder to uncompress the data. Likewise, if the
compressed input was encoded by a JPEG encoder, Yogeshwar's system can *only*

1 use a JPEG decoder to uncompress the data. In the language of Applicant's claim
2 15, Yogeshwar cannot use, *even in part*, a *second decoder* that is *different from*
3 *the first decoder* to uncompress the compressed data. Rather, Yogeshwar can
4 *only* use a *first decoder* that corresponds to the encoder. As such, Yogeshwar
5 teaches *directly away* from the subject matter of this claim. Applicant has
6 reviewed the entire Yogeshwar reference and can find no teaching anywhere of
7 information that is associated with a *second decoder* that is *different from the*
8 *first decoder* and which can be used, at least in part, to *uncompress* the
9 compressed data.

10 Accordingly, for at least these reasons, this claim is allowable.

11 Claims 16-20 depend from claim 15 and, as such, are allowable as
12 depending from an allowable base claim. These claims are also allowable for
13 their own recited features which, in combination with those recited in claim 15,
14 are neither shown nor suggested by the references as cited and applied by the
15 Office.

16
17 Claims 21-22

18 Claim 21 recites a software application comprising [emphasis added]:

- 19
- 20 • an encoding application configured to:
 - 21 ○ receive a file comprising compressed data and information
 - 22 associated with an encoder that compressed source data
 - 23 corresponding to the compressed data, said information being
 - 24 configured for use in locating a first decoder that corresponds to
 - 25 the encoder and which can be used to uncompress the
 - compressed data;
 - search for the information; and
 - replace the information with different information that is
 - associated with a *second decoder* that is *different from the first*

1 ***decoder*** and which ***can be used***, at least in part, to ***uncompress***
2 the compressed data.

3 In making out the rejection of this claim, the Office again argues that
4 Yogeshwar teaches the subject matter of this claim. In response to Applicant's
5 request for clarification from the Office, the Office cited to paragraph 143 to
6 support its argument. The cited excerpt, discussing Yogeshwar's Fig 6, is
7 reproduced above.

8 Applicant will again focus its attention on the ***decoding*** aspect of the cited
9 excerpt. Yogeshwar's "multi-format decoder" shown in Fig 6 performs the partial
10 decoding referenced above. The multi-format decoder in Fig 6 apparently has the
11 same structure as the multi-format decoder in Fig 5. Additional information
12 regarding the multi-format decoder in Fig 5 can be found in paragraph 107, also
13 reproduced above.

14 According to the excerpt disclosing the multi-format decoder, if the
15 compressed input was encoded by an MPEG encoder, for example, Yogeshwar's
16 system can ***only*** use an MPEG decoder to uncompress the data. Likewise, if the
17 compressed input was encoded by a JPEG encoder, Yogeshwar's system can ***only***
18 use a JPEG decoder to uncompress the data. In the language of Applicant's claim
19 21, Yogeshwar cannot use, ***even in part***, a ***second decoder*** that is ***different from***
20 ***the first decoder*** to ***uncompress*** the compressed data. Rather, Yogeshwar can
21 ***only*** use a ***first decoder*** that corresponds to the encoder. As such, Yogeshwar
22 teaches ***directly away*** from the subject matter of this claim. Applicant has
23 reviewed the entire Yogeshwar reference and can find no teaching anywhere of
24 information that is associated with a ***second decoder*** that is ***different from the***
25

1 *first decoder* and which can be used, at least in part, to *uncompress* the
2 compressed data.

3 Accordingly, for at least these reasons, this claim is allowable.

4 **Claim 22** depends from claim 21 and, as such, is allowable as depending
5 from an allowable base claim. This claim is also allowable for its own recited
6 features which, in combination with those recited in claim 21, are neither shown
7 nor suggested by the references as cited and applied by the Office.

8
9 **Claims 23-28**

10 **Claim 23** recites a *decoder application* comprising a *wrapper for a first*
11 *decoder* that is associated with an encoder that was used to compress original
12 source data, the wrapper being configured to [emphasis added]:

- 13
- 14 • *receive compressed source data* from a rendering application;
 - 15 • *provide the compressed source data to the first decoder* so that the
16 compressed source data can be uncompressed;
 - 17 • *receive back modified source data* that has been modified in some
18 way so that the modified source data is different from the original
19 source data; and
 - 20 • *provide the modified source data* to the rendering application for
21 rendering.

22 In making out the rejection of this claim, the Office once more argues that
23 Yogeshwar teaches the subject matter of this claim. In the Office Action, the
24 Office cites to paragraphs 50-71 to support its argument. The cited excerpt is
25 reproduced below [emphasis added]:

23 [0059] *Encoding format/level selection* can be done base on an
24 analysis of needs and features, or as a look-up in predefined tables,
25 or as a learning process. The transcoder, discussed below, may
include similar functionality for making transcoding format/quality

level *selections* and/or suggestions.

[0060] In the exemplary capture and compression module discussed below, an *encoding format and/or encoding parameters* appropriate for the best of a number of anticipated applications, subject to a minimum threshold of high quality suitable for an archive may automatically be *determined*.

[0061] Some exemplary pairings between use or application (on the left) as specified by received input or data analysis and the *selected or suggested video encoding format/level* (on the right) are:

[0062] Digital TV distribution use => MPEG-2 MP@ML
format/level

[0063] Digital TV production use => MPEG-2 422P@ML

[0064] High definition TV => a high definition mode of MPEG-2

[0065] Medical applications => lossless or near-loss encoding

[0066] Head-and-shoulders, videoconference, surveillance, etc, =>
MPEG-1 or H.263

[0067] Browsing delivery by transcoder => a browsing format such as Windows Media, Real, or QuickTime with bitrate set according to user's capabilities and bandwidth

[0068] Wireless delivery => MPEG-4

[0069] Information on previous encoding formats and/or data storage medium used to store the information being processed may also be considered and used when making an *encoding format selection* in accordance with the present invention.

[0070] As mentioned above, analysis of the information to be encoded may be performed as part of the *encoding format selection* process. Information generated as part of the analysis operation may include encoding complexity information expressed in terms of a number of different levels, motion estimates between frames, format and/or content analysis. Some or all of this information may be used as metadata which can be added, e.g., tagged, to the encoded data created as part of the encoding process.

1 [0071] A capture and compression operation is performed on the
2 received information to be archived in step 108. The capture and
3 compression operation 108 involves an A/D conversion in the case
4 of analog input. Digital input or digital data produced by an A/D
5 conversion operation is then followed by an encoding process using
6 the *encoding format/level* determined in step 107. As part of the
7 encoding process, the digital data is encoded according to a format
8 and to a quality level *determined as a function of various input*
9 *information* and/or information generated by analyzing the data to
10 be archived. The encoding operation produces encoded digital data
11 109 which is in an IAF encoding format.

12 As an initial matter, Applicant claims a *wrapper* for a first decoder. Earlier
13 in this response, Applicant provides information that further describes the
14 meaning of the term "wrapper." Applicant has reviewed the Yogeshwar reference
15 and can find no teaching of *wrapper* for a first decoder, as Applicant defines and
16 uses the term. Because Yogeshwar does not teach a wrapper, as utilized in this
17 claim, Yogeshwar cannot possibly teach a wrapper that is configured to *provide*
18 *compressed source data* to a first decoder and *receive back modified source data*
19 that is different from the original source data.

20 Accordingly, for at least these reasons, this claim is allowable.

21 Claims 24-28 depend from claim 23 and, as such, are allowable as
22 depending from an allowable base claim. These claims are also allowable for
23 their own recited features which, in combination with those recited in claim 23,
24 are neither shown nor suggested by the references as cited and applied by the
25 Office.

Conclusion

Applicant respectfully submits that all claims are in condition for allowance. Accordingly, Applicant requests that a Notice of Allowability be issued. If the Office's next anticipated action is to be anything other than issuance of a Notice of Allowability, Applicant requests that the undersigned be contacted for the purpose of scheduling an interview.

Respectfully submitted,

Dated: 9/21/04

By: RR Cottle
Rob R. Cottle.
Reg. No. 52,772
(509) 324-9256 ext. 247